

**IN THE SPECIFICATION:**

Please replace the full paragraph of specification spanning pages 3-4 with the following replacement paragraph:

— On a different note, one known traffic-policing algorithm is the “leaky bucket” algorithm (for example as used in the Asynchronous Transfer Mode (ATM) Protocol, and as described in the ATM Forum’s Traffic Management Specification Version 4.1). ATMs forward fixed size packets known as “cells.” A continuous-state leaky bucket algorithm, as its name implies, can be imagined as a finite-capacity bucket (actually a queue or a counter) in which a real-valued content drains out at a continuous rate of 1 unit of content per time-unit and whose content is increased by the increment I for each conforming cell. The leaky bucket algorithm is fully described by Andrew S. Tanenbaum in his book *Computer Networks, Third Edition*, published by Prentice Hall, Copyright date 1996, all disclosures of which are incorporated herein by referenced, particularly at pages 380-381. As shown in Fig. 1, at block 100, the algorithm is activated when a cell is received. At its initiating state, the content of the bucket is zero. With the arrival of the first cell ta(1), the Last Conformance Time (LCT) is set to ta(1). With the successive arrival of the cells such as the kth cell at time ta(k), at block 102, the content of the bucket x’ is updated to equal to the value of the leaky bucket at the arrival of the last conforming cell minus the amount the bucket has drained since that arrival. Note that the content of the bucket can-not be less than zero and at blocks 104-106, if the content of the bucket x’ is less than zero, the value x’ is adjusted to zero. At block 108, if the value x’ is greater than a limit value L, the cell is non-conforming and at block 110, the values of x’ and LCT remain unchanged. Otherwise if the value x’ is less than or equal to the limit value L, the cell is conforming and at block 112, the bucket content x is set to x’ plus the increment I for that current cell and the LCT is set to ta(k). Further details may be found in the forum paper specified above. —